2.5 DETAILED RESULTS FOR ONBOARD NOISE ECM

Several discrepancies were found in the Onboard Noise ECM Functional Element (FE) for ESAMS 2.7. The overall code quality is good; however, numerous corrections are recommended for the internal documentation. External documentation for ESAMS 2.7 is generally good although some design elements had errors or were incomplete in their discussion and presentation of ECM theory and methodology.

Table 2.5-1 summarizes the desk-checking and software testing verification activities for each design element in the Onboard Noise ECM FE. A design element is an algorithm that represents a specific component of the FE design. One entry is listed for each design element. The two results columns contain checks if no discrepancies were found during verification. Where discrepancies were found, the desk check results column contains references to discrepancies listed in Table 2.5-4, while the test case results column lists the number of the relevant test case in Table 2.5-6. More detailed information on the results is recorded in these tables.

TABLE 2.5-1. Verification Results Summary.

DESIGN ELEMENT	CODE LOCATION	DESK CHECK RESULT	TEST CASE ID	TEST CASE RESULT
5-1 Geometry	BEMTVL 82-97 BEMSVL 145-162 BEMFVL 93-110 BEMOUT 126-156	D1 D2 D3	5-1 to 5-3	5-1 to 5-3
5-2 Doppler	BEMTVL 102 BEMSVL 179 BEMFVL 113	D4	5-4 to 5-6	5-6
5-3 Phase	BEMTVL 106 BEMSVL 186 BEMFVL 122-123	4	5-4 to 5-6	5-6
5-4 Power	BEMTVL 104, 108-110 BEMSVL 181-184, 188-190 BEMFVL 118, 120, 125-127	4	5-4 to 5-6	5-6
5-5 Jammer Sensed Voltages and Gains	Not Found	D5	n/a	n/a

TABLE 2.5-1. Verification Results Summary. (Contd.)

DESIGN ELEMENT	CODE LOCATION	DESK CHECK RESULT	TEST CASE ID	TEST CASE RESULT
5-6 Jamming Waveform for Repeater (Relative) Jammers	BEMEXC 56-101	D6 D7	5-7	4
5-7 Complex Waveforms at Radar	BEMOUT 157-188	D8	5-8	5-8
5-8 Adjust Power Levels	BEMGRM 427-446	4	5-9	4
5-9 Continuous Noise Amplitude Calculation	BEMOUT 157-176 BEMNZ 139-169	D9	5-10 5-11	5-10

2.5.1 Overview

The objective of noise jamming is to inject an interference signal into the enemy's radar such that the target return signal is completely submerged by the interference. This is accomplished by the jamming radar's noise power exceeding the power of the pulse returned to the victim radar, thus the signal reflected from the intended target gets lost in the jamming noise. Onboard Noise ECM involves the generation of noise-like jamming signals from sources originating on the intended target itself as opposed to those apparently originating from off-board locations. There are three different types of noise jamming: spot jamming, broad-band barrage jamming, and swept spot noise (SSN) jamming, with the third type being a combination of the first two. The Onboard Noise ECM Functional Element for ESAMS models the second and third types of noise jamming with the latter type implemented using two different techniques, bin masking and SSN.

ESAMS 2.7 implementation of Onboard Noise ECM is accomplished with eleven primary subroutines. Subroutines BEMGRM, BEMSEN, BEMTVL, BEMSVL, BEMFVL, BEMANT, BEMEXC, BEMOUT, BEMNZ and BEMSET are the primary users for the Onboard Noise ECM FE. In addition, ECMINI is a higher level subroutine that is used to initialize ECM techniques. None of these subroutines are exclusively designated for this FE and are shared with one or more of the other FE's that model ECM techniques. However, these subroutines are exclusively designated for ECM techniques in general, with Onboard Noise being one of three ECM FE categories modeled by ESAMS. The eleven subroutines used for this FE are described in Table 2.5-2.

TABLE 2.5-2. Onboard Noise ECM Subroutine Descriptions.

MODULE NAME	DESCRIPTION
BEMGRM	Checks each technique in the ECMD file to see if it is active at the current time against the current radar. Serves as top level routine for ECM calculations.
BEMSEN	Sets up engagement features between the jamming aircraft and the ground radar, missile seeker, or missile fuze.
BEMTVL	Calculates relative geometries and orientations between the ground radar and jamming aircraft.

TABLE 2.5-2. Onboard Noise ECM Subroutine Descriptions. (Contd.)

MODULE NAME	DESCRIPTION
BEMSVL	Calculates relative geometries and orientations between the missile seeker and jamming aircraft.
BEMFVL	Calculates relative geometries and orientations between the missile fuze and jamming aircraft. Obtains fuze characteristics.
BEMANT	Provides jamming antenna position, velocity, and orientation.
BEMEXC	Loads the current ECM characteristics. These include Doppler, power, phase, polarization, pulse width, and time delay.
BEMOUT	Develops the ECM-induced voltage in the victim radar receiver for bin masking and swept spot noise.
BEMNZ	Computes the noise-like jamming signals for continuous noise ECM.
BEMSET	Sets flags for printing event message output. Each ECM signal is examined for turning on or off.
ECMINI	Initializes ECM simulations. Provides initialization for both noise and ECM techniques.

2.5.2 Verification Design Elements

Design elements defined for the Onboard Noise ECM FE are listed in Table 2.5-3; they are fully described in Section 2.5.2 of the ASP II. A design element is an algorithm that represents a specific component of the FE design. Design elements 5-1 through 5-9 model the functions used to implement Onboard Noise ECM in ESAMS 2.7.

TABLE 2.5-3. Onboard Noise ECM Design Elements.

SUBROUTINE	DESIGN ELEMENT	DESCRIPTION
BEMTVL BEMSVL BEMFVL BEMOUT	5-1 Geometry	Calculates the range, relative velocities, and jammer antenna pointing angles between the jammer and victim radar. The range and velocities are used to calculate returned Doppler, power, and phase.
BEMTVL BEMSVL BEMFVL	5-2 Doppler	Calculates the Doppler shift due to the relative velocity between the jammer and the victim radar.
BEMTVL BEMSVL BEMFVL	5-3 Phase	Calculates the phase at the jammer due to the slant range from the jammer to the victim radar.
BEMTVL BEMSVL BEMFVL	5-4 Power	Calculates both the power density at the target and the power sensed by the jammer due to the threat radar transmitter.
Not Found	5-5 Jammer Sensed Voltages and Gains	This design element doesn't make sense. These equations appear to be for a monopulse radar, not a jamming radar. Couldn't find them implemented in any of the subroutines making up this FE.
BEMEXC	5-6 Jamming Waveform for Repeater (Relative) Jammers	Calculates the waveform transmitted to the victim radar for repeater-type jammers. Used when jammer characteristics are determined relative to the victim radar.

TABLE 2.5-3. Onboard Noise ECM Design Elements. (Contd.)

SUBROUTINE	DESIGN ELEMENT	DESCRIPTION
BEMOUT	5-7 Complex Waveforms at Radar	Calculates the complex waveforms at the victim radar. The complex voltage is determined in the receiver's sum, difference elevation, and difference azimuth channels.
BEMGRM	5-8 Adjust Power Levels	Adjusts jammer-induced voltages in the victim radar receiver to prevent the total jammer power calculated from exceeding the maximum jammer power available. The voltages in the receiver are reduced in proportion to the square of the maximum jammer power-sum of all jammer technique powers ratio if any adjustments are necessary.
BEMOUT BEMNZ	5-9 Continuous Noise Amplitude Calculation	The most basic of ECM techniques, it simulates the propagation of broad band noise, and is meant to fill up the entire PRI of the victim radar. Returns sum and difference channel voltages in the victim radar receiver due to continuous noise ECM. Impacts a system's target acquisition and tracking capability.

2.5.3 Desk Checking Activities and Results

The code implementing this FE was manually examined using the procedures described in Section 1.1 of this report. Any discrepancies discovered are described below in Table 2.5-4.

TABLE 2.5-4. Code Discrepancies.

DESIGN ELEMENT	DESK CHECK RESULT
5-1 Geometry	D1. The position components for the jammer and victim radars have been transposed in the X, Y, and Z range calculations for ASP II Equation [2.5-1] with respect to what has been implemented in the subroutines BEMTVL, BEMSVL, and BEMFVL. D2. The velocity components for the jammer and victim radars have been transposed in the X, Y, and Z separation rate calculations for ASP II Equation [2.5-2] with respect to what has been implemented in the subroutines BEMTVL, BEMSVL, and BEMFVL. D3. For the case of a slewable antenna, the off-boresight azimuth angle, AZ2V, is set to zero on the consecutive lines 153 and 154 in the subroutine BEMOUT. Since the comments imply that there are at least two angles to be set to zero, one of these lines should be changed to the off-boresight elevation angle, EL2V.
5-2 Doppler	D4. The calculation of RADVLU(1) on line 113 of the subroutine BEMFVL for Doppler shift uses the wavelength of the illuminator radar, WVLTX(4), while the calculations for other elements in the RADVLU array use the wavelength of the fuze radar, WVLFUZ. Since the values calculated in this array should be for characteristics at the target with respect to the victim radar, the equation for Doppler shift should use the wavelength of the fuze radar rather than that of the illuminator.

TABLE 2.5-4. Code Discrepancies. (Contd.)

DESIGN ELEMENT	DESK CHECK RESULT
5-5 Jammer Sensed Voltages and Gains	D5. The ASP II Equation(s) [2.5-8] were not found to be implemented in any of the subroutines listed for this FE. These equations are for sum and difference channels which are commonly found in monopulse radars, but not in jammer radars. This design element should be removed from Section 2.5 of the ASP II.
5-6 Jamming Waveform for Repeater (Relative) Jammers	D6. The equation to calculate the bistatic RCS value used in ASP II Equation [2.5-10] is inappropriate. It is calculated in the subroutines GETRCS and TGTRCS by taking the geometric mean of the monostatic RCS as observed from the transmitter location and that observed from the receiver location. The only case in which this would be correct would be for a bistatic angle of 0 degrees, i.e., a monostatic case. The developer should look at alternative ways of approximating this quantity. D7. The equation for the array element VALUE(4) should be included in this design element but was omitted. Independent review found the implementation of this equation on line 92 of the subroutine BEMEXC to be questionable. Since the comments in the subroutine state that it is for the case of polarization, it appears that a polarization factor is taken from an input ECM table and added to the power sensed by the jammer. If the equation is to account for polarization mismatch between the jammer-victim radar antennas, then this polarization should be multiplied by the sensed power, not added to it. Otherwise, it is not clear what this equation is trying to accomplish; thus, it should be reviewed by the developer for correct implementation.
5-7 Complex Waveforms at Radar	D8. The equations to adjust the difference channel complex voltages for cross-polarization bleedover are not explicitly shown in this design element, but nevertheless, can be considered part of it. An independent review found the equations implemented on lines 190 and 192 of the subroutine BEMOUT to be incorrect. The second term in these equations has a power being multiplied by a voltage which would result in a voltage cubed. This, in turn, would be added to the voltage in the first term which doesn't make sense. These equations should be reviewed by the developer and corrected as necessary.
5-9 Continuous Noise Amplitude Calculation	D9. The equations for implementing the complex voltage in the subroutines BEMOUT and BEMNZ do not agree with the ASP II Equation [2.5-18] (unnumbered in the ASP II). The equations in the code appear to be correct, with the errors being in the ASP II. The voltages $V_{\Sigma J}$, V_{-aJ}, and V_{-eJ} on the right-hand side of this equation(s) appear to be the complex voltages calculated in Design Element 5-7. If that is the case, then they include the complex phase angle calculated in ASP II Equation [2.5-14] which is not used in the calculation of voltages for continuous noise jamming. These voltages should be the same as the voltages calculated in the ASP II Equation [2.5-15], but with the phase term from the ASP II Equation [2.5-14] removed. This appears to be primarily a notation error, but should be corrected to avoid confusion. The notation used for these variables is also identical with that used in Design Element 5-5, which could cause further confusion.

Except as noted in Table 2.5-5 below, overall code quality was evaluated as good and internal documentation was evaluated as fair. In most cases, subroutine I/O and logical flow were found to match the ASP II descriptions. However, significant flowchart errors were found for the subroutine BEMGRM and minors ones were found for most of the other subroutines.

TABLE 2.5-5. Code Quality and Internal Documentation Results.

SUBROUTINE	CODE QUALITY	INTERNAL DOCUMENTATION
BEMGRM	The INCLUDE statements on lines 212 and 220 for the common blocks ECCHAF and PROGC are not necessary since the variables they contain are not used.	The definition of the local variable PSIBM is wrong. It should be defined as the target yaw angle, not the target roll angle. The variable RADCHF is missing from the list of local variables.
BEMSEN	OK	The definitions of the calling arguments XVJ, YVJ, and ZVJ are wrong. They should be defined as the X, Y, and Z components of the victim site-to-jammer antenna vector. Definitions for ANTPHI, ANTTHT, and ANTPSI are missing from the list of local variables. The variables AMISX, AMISY, AMISZ, FUZX, FUZY, FUZZ, XSJ, YSJ, and ZSJ are in this list but are not used. The subroutines AFMPOS, MISPOS, and SITPOS are in the list of subroutines called by BEMSEN but are not used.
BEMTVL	The INCLUDE statement on line 71 for the common block FREND is not necessary since the variables that it contains are not used.	The variables ANTPHI, ANTTHT, and ANTPSI are missing from the list of calling arguments. The variables ANULL and RGAIN are missing from the list of local variables. The variable ILUMR is missing from the list of parameters. The subroutine ANTGAN is missing from the list of subroutines called by BEMTVL. The comment on line 103 for the variable RADVLU(2) is incorrect. This equation is actually for the power density at the target.

TABLE 2.5-5. Code Quality and Internal Documentation Results. (Contd.)

SUBROUTINE	CODE QUALITY	INTERNAL DOCUMENTATION
BEMSVL	The INCLUDE statements on lines 126 and 127 for the common blocks FLAGS and FREND are not necessary since the variables they contain are not used.	The variables ANTPHI, ANTTHT, and ANTPSI are missing from the list of calling arguments. The variables ANULL and RGAIN are missing from the list of local variables. The variables RANGE, XAT, YAT, ZAT, XSAT, YSAT, and ZSAT are in this list but are not used. FLAGS and FREND are in the list of common blocks but the variables they contain are not used. The variable ALPOFF is in the list of variables for the common block GRADAR but is not used. The subroutine ANTGAN is missing from the list of subroutines called by BEMSVL. The comment on line 183 for the variable RADVLU(2) is incorrect. This equation is actually for the power density at the target.
BEMFVL	The INCLUDE statement on line 79 for the common block FREND is not necessary since the variables that it contains are not used.	The variables ANTPHI, ANTTHT, and ANTPSI are missing from the list of calling arguments. The variables ANULL and RGAIN are missing from the list of local variables. The variables OLDRCS and OLDTIM are in this list but are not used. The common blocks ARYBND and FLAGS with their respective variables IRADFL and NRCHAR are missing from the list of common blocks. The subroutines ANTGAN, FEND, and MISVEL are missing from the list of subroutines called by BEMFVL. The subroutines AFMFND, AFMVEL, and TGTROL are in the list of subroutines called but are not used. The comment on line 119 for the variable RADVLU(2) is incorrect. This equation is actually for the power density at the target.
BEMANT	INCLUDE statements for ARYBND and PARAM are not necessary since the parameters contained in these common blocks are not used. Missing variable declaration statements for IANT, IONDCY, ITCHNQ, TIMEB, ANTX, ANTY, ANTZ, ANTXD, ANTYD, ANTZD, ANTPSI, ANTTHT, ANTPHI, TGTPSI, TGTTHT, TGTPHI, TGTX, TGTY, TGTZ, TGTXD, TGTYD and TGTZD need to be added to the code.	The variables ANTPHI, ANTTHT, and ANTPSI are missing from the list of calling arguments. The library function NINT and the subroutine TDROLL are missing from the list of subroutines called by BEMANT.

TABLE 2.5-5. Code Quality and Internal Documentation Results. (Contd.)

SUBROUTINE	CODE QUALITY	INTERNAL DOCUMENTATION
BEMEXC	OK	The variable IONFLG is missing from the list of local variables. The library functions NINT and FLOAT along with the subroutine GETWOB are missing from the list of subroutines called by BEMEXC.
BEMOUT	The calls to the subroutine GYRATE on lines 137 and 147 are redundant and can be consolidated into one call. This can be accomplished by moving the call to line 131 so that it occurs before the IF statement for the fixed or slewable antenna. The functions ASIN and ATAN2 on lines 141 and 143, respectively, can be replaced by the functions ASIND and ATAND2 which return answers in degrees, not radians. This would eliminate the need to convert them to degrees by multiplying them by the radians-to-degrees conversion factor R2D.	Comments for the calculation of the variables ECHVLT and ACHVLT state that these are difference channel gains, when in fact they are difference channel voltages. The variables IDBUS, IXPNT1, IXPNT2, KODAMP, NUMPRO, PCBWJM, SPCWID, and XMTPAT are missing from the list of calling arguments. The variables I2B, KFREQ, KPWR, KPHASE, KPWID, KRVGAZ, KRVGEL, KRVSUM, KRVVCL, and KTDEL are missing from the list of parameters. The variable PIX2 is in this list but is not used. The subroutines BEMNZ and TGTROL are missing from the list of subroutines called by BEMOUT. The library function AMOD is in this list but is not used.
BEMSET	OK	The variable NUMTEC is in the calling argument list for BEMSET but is not used.
ECMINI	INCLUDE statements for the common blocks FLAGS, GRADAR, and PROGC are not necessary because the variables contained in them are not used.	The subroutine contains no version number. The source code shows this subroutine to be ‘UNCLASSIFIED’, when in fact it is actually classified as ‘SECRET’. The variables IERRIP, IPT, JPT, ISPS, ISPS5, MODE, NUMWDS, and PATFIL are missing from the list of local variables. The indices I and J are in this list but are not used. The variables IFATAL and IWARN are missing from the parameters list. The variables IACQR, ILUMR, ISEKR, ITRKR, MRADFL, and NUMJAM are in this list but are not used. The variables ANSLW, CHRPT, CNTFRQ, DUTCYL, ECMT, LPATRN, OFFFRQ, PANT, RMPTIM and SWPTYP are missing from the list of variables for the common block ECMD.

TABLE 2.5-5. Code Quality and Internal Documentation Results. (Contd.)

SUBROUTINE	CODE QUALITY	INTERNAL DOCUMENTATION
ECMINI (Continued)		JCHFRC is in this list of variables for the common block ECMI but is not used. GRADAR and PROGC are in the list of common blocks but their respective variables WVLTX and CHFRC are not used. The common blocks ECMV, PROGVI and RUNVI with their respective variables TIMMOD, ISPS(5) and LUNLP are missing from the list of common blocks. The subroutines CKTLU2, ERROR, and RDF along with the library function NINT are missing from the list of subroutines called. The subroutine CHAFFI is in this list but is not used. The comments on lines 140, 145, and 160 that refer to the transmit antenna are incorrect. These comments should be referring to the receive antenna instead.

2.5.4 Software Test Cases and Results

With only one exception, all subroutines implementing the Onboard Noise ECM FE were tested using integrated code. For test case 5-11, off-line testing of a uniform random number generator was performed using a modified copy of the code for the subroutine UNIRAN on a VAXstation 3100. For integrated testing, the entire ESAMS model was run in debug mode. The standard ESAMS data files for the systems under consideration were used as input for all test cases, except for the one mentioned above. In another case, a jammer data file contained errors which needed correcting before verification activities could be completed.

TABLE 2.5-6. Software Test Cases for Onboard Noise ECM FE.

Test Case ID	Test Case Description
5-1	OBJECTIVE: Verify correct calculation of range, relative velocity, and off-boresight angles between jammer and tracking radar for case of fixed jammer antenna. PROCEDURE: 1. Run ESAMS, and observe in Subroutine BEMSEN the execution path following line 78. 2. Observe in Subroutine BEMTVL the values of XJST, YJST, ZJST, and RJST. 3. Observe in Subroutine BEMTVL the values of XJSTD, YJSTD, ZJSTD, and VCLJ. 4. Continue execution and observe in Subroutine BEMOUT the execution path following line 132. 5. Observe in Subroutine BEMOUT the values of EL2V and AZ2V. VERIFY: 1. Execution transfers to line 81 in step 1. 2. The value of RJST observed in step 2 matches independent calculation of ASP II Equation [2.5-1] where the variables XJST, YJST, and ZJST correspond to the ASP equation variables X_{JR}, Y_{JR}, and $Z_{JR,}$ respectively. 3. The value of VCLJ observed in step 3 matches independent calculation of ASP II Equation [2.5-2] where the variables XJSTD, YJSTD, and ZJSTD correspond to the ASP equation variables XV_{JR}, YV_{JR}, and ZV_{JR}, respectively. 4. Execution transfers to line 134 in step 4. 5. The values observed in step 5 match independent calculation of ASP II Equation [2.5-3] where the variable EL2V corresponds to the first of the two ASP equations for $ANGLE_e$. RESULT: 1. Hand calculations for X_{JR}, Y_{JR}, and Z_{JR} in step 2 result in values that are equal to but have the opposite sign from those observed in the code. This is because the jammer and victim radar position components are transposed in the ASP II equations with respect to what was implemented in BEMTVL. Despite this discrepancy the calculation of the slant range, RJST agrees with ASP II Equation [2.5-1]. 2. The same problem exists for XV_{JR}, YV_{JR}, and ZV_{JR} in step 3, the hand calculated values are equal to but have the opposite sign from those observed in the code. Like before, this is because the jammer and victim radar velocity components are transposed in the ASP II equations with respect to what was implemented in BEMTVL. However, these component sign errors will cancel each other out in ASP II Equation [2.5-2], resulting in the correct calculation of VCLJ. Otherwise OK.

TABLE 2.5-6. Software Test Cases for Onboard Noise ECM FE. (Contd.)

Test Case ID	Test Case Description
5-2	OBJECTIVE: Verify correct calculation of range, relative velocity, and off-boresight angles between jammer and seeker radar for case of fixed jammer antenna. PROCEDURE: 1. Run ESAMS, and observe in Subroutine BEMSEN the execution path following line 78. 2. Observe in Subroutine BEMSVL the values of XJST, YJST, ZJST, and RJST. 3. Observe in Subroutine BEMSVL the values of XJSTD, YJSTD, ZJSTD, and VCLJ. 4. Continue execution and observe in Subroutine BEMOUT the execution path following line 132. 5. Observe in Subroutine BEMOUT the values of EL2V and AZ2V. VERIFY: 1. Execution transfers to line 86 in step 1. 2. The value of RJST observed in step 2 matches independent calculation of ASP II Equation [2.5-1] where the variables XJST, YJST, and ZJST correspond to the ASP equation variables X_{JR}, Y_{JR}, and $Z_{JR,}$ respectively. 3. The value of VCLJ observed in step 3 matches independent calculation of ASP II Equation [2.5-2] where the variables XJSTD, YJSTD, and ZJSTD correspond to the ASP equation variables XV_{JR}, YV_{JR}, and $ZV_{JR,}$ respectively. 4. Execution transfers to line 134 in step 4. 5. The values observed in step 5 match independent calculation of ASP II Equation [2.5-3] where the variable EL2V corresponds to the first of the two ASP equations for $ANGLE_e$. RESULT: 1. Hand calculations for X_{JR}, Y_{JR}, and Z_{JR} in step 2 result in values that are equal to but have the opposite sign from those observed in the code. This is because the jammer and victim radar position components are transposed in the ASP II equations with respect to what was implemented in BEMSVL. Despite this discrepancy the calculation of the slant range, RJST agrees with ASP II Equation [2.5-1]. 2. The same problem exists for XV_{JR}, YV_{JR}, and ZV_{JR} in step 3, their hand-calculated values are equal to but have the opposite sign from those observed in the code. Like before, this is because the jammer and victim radar velocity components are transposed in the ASP II equations with respect to what was implemented in BEMSVL. However, these component sign errors will cancel each other out in ASP II Equation [2.5-2], resulting in the correct calculation of VCLJ. Otherwise OK.

TABLE 2.5-6. Software Test Cases for Onboard Noise ECM FE. (Contd.)

Test Case ID	Test Case Description
5-3	OBJECTIVE: Verify correct calculation of range, relative velocity, and jammer antenna pointing angles between jammer and fuze radar for case of slewable jammer antenna. PROCEDURE: 1. Run ESAMS, and observe in Subroutine BEMSEN the execution path following line 89. 2. Observe in Subroutine BEMFVL the values of XJST, YJST, ZJST, and RJST. 3. Observe in Subroutine BEMFVL the values of XJSTD, YJSTD, ZJSTD, and VCLJ. 4. Continue execution and observe in Subroutine BEMOUT the execution path following line 132. 5. Observe in Subroutine BEMOUT the values of APOYNT and EPOYNT. VERIFY: 1. Execution transfers to line 91 in step 1. 2. The value of RJST observed in step 2 matches independent calculation of ASP II Equation [2.5-1] where the variables XJST, YJST, and ZJST correspond to the ASP equation variables X_{JR}, Y_{JR}, and $Z_{JR,}$ respectively. 3. The value of VCLJ observed in step 3 matches independent calculation of ASP II Equation [2.5-2] where the variables XJSTD, YJSTD, and ZJSTD correspond to the ASP equation variables XV_{JR}, YV_{JR}, and $ZV_{JR,}$ respectively. 4. Execution transfers to line 147 in step 4. 5. The values observed in step 5 match independent calculation of ASP II Equation [2.5-3] where the variable EPOYNT corresponds to the second of the two ASP equations for $ANGLE_e$. RESULT: 1. Hand calculations for X_{JR}, Y_{JR}, and Z_{JR} in step 2 result in values that are equal to but have the opposite sign from those observed in the code. This is because the jammer and victim radar position components are transposed in the ASP II equations with respect to what was implemented in BEMFVL. Despite this discrepancy the calculation of the slant range, RJST agrees with ASP II Equation [2.5-1]. 2. The same problem exists for XV_{JR}, YV_{JR}, and ZV_{JR} in step 3, their hand-calculated values are equal to but have the opposite sign from those observed in the code. Like before, this is because the jammer and victim radar velocity components are transposed in the ASP II equations with respect to what was implemented in BEMFVL. However, these component sign errors will cancel each other out in ASP II Equation [2.5-2], resulting in the correct calculation of VCLJ. 3. The equations for APOYNT and EPOYNT in step 5 are actually for the antenna pointing angles to the victim radar for the case of a slewable antenna. Therefore, the angles off-boresight, EL2V and AZ2V, are set to zero (this needs to be corrected in the code since AZ2V was set to zero twice and EL2V was omitted). The equations in the ASP II are correct, its just that these variables are defined wrong in this case. The ASP II variables $ANGLE_e$ and $ANGLE_a$ should be renamed to something unique in this case to avoid confusion with the off-boresight angles calculated for the case of a fixed antenna. Otherwise OK.

TABLE 2.5-6. Software Test Cases for Onboard Noise ECM FE. (Contd.)

Test Case ID	Test Case Description
5-4	OBJECTIVE: Verify correct calculation of Doppler shift, phase, and power at the jammer due to the victim radar for case of tracker radar jamming. PROCEDURE: 1. Start ESAMS and observe in Subroutine BEMTVL the values of RJST and VCLJ. 2. Observe in Subroutine BEMTVL the value of RADVLU(1). 3. Observe in Subroutine BEMTVL the value of RADVLU(2). 4. Observe in Subroutine BEMTVL the value of RADVLU(3). 5. Observe in Subroutine BEMTVL the value of RADVLU(4). VERIFY: 1. The values observed in step 1 match independent calculations. 2. The value observed in step 2 matches independent calculation of ASP II Equation [2.5-4]. 3. The value observed in step 3 matches independent calculation of ASP II Equation [2.5-6]. 4. The value observed in step 4 matches independent calculation of ASP II Equation [2.5-5]. 5. The value observed in step 5 matches independent calculation of ASP II Equation [2.5-7]. RESULTS: OK
5-5	OBJECTIVE: Verify correct calculation of Doppler shift, phase, and power at the jammer due to illuminator radar for case of seeker radar jamming. PROCEDURE: 1. Observe in Subroutine BEMSVL the value of RJST and VCLJ. 2. Observe in Subroutine BEMSVL the value of RADVLU(1). 3. Observe in Subroutine BEMSVL the value of RADVLU(2). 4. Observe in Subroutine BEMSVL the value of RADVLU(3). 5. Observe in Subroutine BEMSVL the value of RADVLU(4). VERIFY: 1. The values observed in step 1 match independent calculations. 2. The value observed in step 2 matches independent calculation of ASP II Equation [2.5-4]. 3. The value observed in step 3 matches independent calculation of ASP II Equation [2.5-6]. 4. The value observed in step 4 matches independent calculation of ASP II Equation [2.5-5]. 5. The value observed in step 5 matches independent calculation of ASP II Equation [2.5-7]. RESULTS: OK

TABLE 2.5-6. Software Test Cases for Onboard Noise ECM FE. (Contd.)

Test Case ID	Test Case Description
5-6	OBJECTIVE: Verify correct calculation of Doppler shift, phase, and power at the jammer due to victim radar for case of jamming fuze radar using continuous noise ECM technique. PROCEDURE: 1. Observe in Subroutine BEMFVL the value of RJST, VCLJ, and WVLTX(4). 2. Observe in Subroutine BEMFVL the value of RADVLU(1). 3. Observe in Subroutine BEMFVL the value of RADVLU(2). 4. Observe in Subroutine BEMFVL the value of RADVLU(3). 5. Observe in Subroutine BEMFVL the value of RADVLU(4). 6. Continue execution until the simulation is complete. VERIFY: 1. The values observed in step 1 match independent calculations. 2. The value observed in step 2 matches independent calculation of ASP II Equation [2.5-4]. 3. The value observed in step 3 matches independent calculation of ASP II Equation [2.5-6]. 4. The value observed in step 4 matches independent calculation of ASP II Equation [2.5-5]. 5. The value observed in step 5 matches independent calculation of ASP II Equation [2.5-7]. 6. The program successfully reaches completion. RESULTS: 1. The Doppler shift calculation in step 2 uses the wavelength of the illuminator radar, WVLTX(4), while the calculations in steps 3 through 5 uses the wavelength of the fuze radar, WVLFUZ. Since calculation of values in the RADVLU array should be for characteristics at the target with respect to the victim radar, the equation in step 2 should use the wavelength of the fuze radar rather than that of the ground illuminator. 2. The program has a fatal error on line 153 of the subroutine BEMNZ and crashes because of a singularity that occurs when it tries to divide by zero during the calculation of NBINS. This happens because the jammer pulsewidth (PWJAM) is set to zero in the ECMT table of the ECMD file for Continuous Noise jamming. Even if a nonzero pulsewidth were defined in the ECMT table, a fatal program error would still occur because NBINS would be zero since the variables TIMINT and RST aren’t defined and default to zero. This is because the preceding IF statement which normally defines these variables doesn’t account for the case of a fuze radar. Either fix BEMNZ to account for a fuze radar or remove the subroutine BEMFVL (and thus the possibility of jamming a fuze radar) from this FE. Since Continuous Noise jamming is apparently not very effective against fuze radars and the other two ECM techniques can’t be used against it either, this option could be removed from this FE without any serious repercussions. Otherwise OK.

TABLE 2.5-6. Software Test Cases for Onboard Noise ECM FE. (Contd.)

Test Case ID	Test Case Description
5-7	OBJECTIVE: Verify correct calculations for loading ECM characteristics in the VALUE array at the victim radar receiver (acquisition radar) for Bin Masking ECM technique. PROCEDURE: 1. Run ESAMS, and observe in Subroutine BEMEXC the value of array element VALUE(1). 2. Observe in Subroutine BEMEXC the value of array element VALUE(2). 3. Observe in Subroutine BEMEXC the value of array element VALUE(3). 4. Observe in Subroutine BEMEXC the value of array element VALUE(5). 5. Observe in Subroutine BEMEXC the value of array element VALUE(6). VERIFY: 1. The value observed in step 1 equals independent calculation of ASP II Equation [2.5-9]. 2. The value observed in step 2 equals independent calculation of ASP II Equation [2.5-10]. 3. The value observed in step 3 equals independent calculation of ASP II Equation [2.5-11]. 4. The value observed in step 4 equals independent calculation of ASP II Equation [2.5-12]. 5. The value observed in step 5 equals independent calculation of ASP II Equation [2.5-13]. RESULT: OK
5-8	OBJECTIVE: Verify correct calculation of the sum and difference channel complex voltages in the victim radar receiver for the Swept Spot Noise ECM technique. PROCEDURE: 1. Run ESAMS, and observe in Subroutine BEMOUT the value of the CPHAZ. 2. Observe in Subroutine BEMGRM the execution path following line 183. 3. Observe in Subroutine BEMOUT the values of SUMJAM, DFJMAZ, and DFJMEL. VERIFY: 1. The value of CPHAZ in step 1 matches independent calculation of ASP II Equation [2.5-14]. 2. Execution transfers to line 185 in step 2. 3. The values of SUMJAM, DFJMAZ, and DFJMEL in step 3 match independent calculation of ASP II Equation [2.5-15]. RESULT: The calculation of the jammer antenna gain, GAINJ, in Subroutine BEMOUT was incorrect due to the improper setup of the antenna pattern table in the file ALQ184.DAT. GAINJ is used to calculate the variables in step 3 and is equivalent to G_J in the ASP II Equation [2.5-15]. The subroutine TLU2, that looks up the gain value from tables contained in ALQxxx.DAT or ECMD.DAT files, requires that the azimuth angles in these tables increase from left to right. The jammer antenna gain pattern tables for all types of jamming contained in the ALQ184.DAT file have the azimuth values decreasing from left to right instead of increasing. Thus when TLU2 tries to look up a gain value it thinks it is beyond the bounds of the table and returns an incorrect value. This file needs to be corrected to have the azimuth angles increase from left to right in order for the subroutine TLU2 to work properly. Otherwise OK.

TABLE 2.5-6. Software Test Cases for Onboard Noise ECM FE. (Contd.)

Test Case ID	Test Case Description
5-9	OBJECTIVE: Verify correct calculation of adjustments to keep the sum and difference channel voltages in the victim radar receiver within the maximum power of the jammer for the Swept Spot Noise ECM technique. PROCEDURE: 1. Run ESAMS, and observe the execution path following line 430 in the Subroutine BEMGRM. 2. Observe in Subroutine BEMGRM the value of the PMRAT. 3. Observe in Subroutine BEMGRM the values of SGSV, SGDVA, and SGDVE. VERIFY: 1. Execution transfers to line 432 in step 1. 2. The value observed in step 2 matches independent calculation of ASP II Equation [2.5-16]. 3. The values observed in step 3 match independent calculation of ASP II Equation [2.5-17]. RESULT: OK
5-10	OBJECTIVE: Verify correct calculation of the sum and difference channel voltages at the victim radar receiver (acquisition radar) for the Continuous Noise ECM technique. PROCEDURE: 1. Run ESAMS, and observe in Subroutine BEMOUT the values of SCHVLT, ACHVLT, and ECHVLT. 2. Observe in Subroutine BEMOUT the execution path following line 183. 3. Observe in Subroutine BEMNZ the values of SGSV, SGDVA, and SGDVE. VERIFY: 1. The values in step 1 match independent calculations. 2. Execution transfers to line 208 in step 2. 3. The values of SGSV, SGDVA, and SGDVE in step 3 match independent calculation of ASP II Equation [2.5-18]. RESULT: Due to the randomness of the values for U_G and U_U contained in ASP II Equation [2.5-18], it was not possible to completely verify the results of this equation with what was observed in the code. In addition, the equations for these random variables were not provided in the ASP II, thus making it difficult to ascertain if they were implemented correctly in the code. Only the non-random variables in this equation, i.e., those observed in step 1, can be verified against the ASP II. See test case 5-11 for a correctness check of the random uniform distribution variable U_U. Otherwise OK.
5-11	OBJECTIVE: Create a driver to verify correct implementation of the uniform random number generator in subroutine UNIRAN that is used with the Continuous Noise ECM technique. PROCEDURE: 1. Add a DO loop to cycle through the random number generation equations in the subroutine UNIRAN 10,000 times, initializing ISEED to 1. 2. Create an output file and add statements to write values of the DO loop index, I and the random number seed, ISEED to this file. 3. Execute the driver (named UNITEST) and observe the value of ISEED after 10,000 DO loop iterations, i.e. I = 10001. VERIFY: 1.The values of ISEED and I observed in step 3 equal 1043618065 and 10001. RESULT: OK

2.5.5 Conclusions and Recommendations

2.5.5.1 Code Discrepancies

In general, the coded algorithms implement the design criteria correctly although several serious discrepancies were uncovered during verification of the Onboard Noise ECM FE for ESAMS 2.7. For the first two discrepancies, position and velocity components have been transposed during the calculation of range and separation rates for X, Y, and Z components. This results in hand calculations using the ASP II equations having values that are equal to, but of opposite sign of what is calculated by the code.

For the third discrepancy, the azimuth off-boresight angle (AZ2V) was set to zero on two consecutive lines in the subroutine BEMOUT. One of these statements should be modified to set the elevation off-boresight angle, EL2V, to zero instead.

For the fourth discrepancy, the wrong wavelength was used for the calculation of Doppler shift due to jammer and victim radar velocities in the case of a fuze radar. The wavelength of the fuze radar should be used instead of that for an illuminator radar as was implemented in BEMFVL.

For the fifth discrepancy, no coding could be found to implement the equations described in Design Element 5-5. This design element should be removed from the ASP II.

For the sixth discrepancy, the equation for the calculation of bistatic RCS from monostatic RCS values is incorrect. Finding the right equation to accomplish this will not be easy and may require rethinking as to how this quantity is to be implemented in the code.

For the seventh discrepancy, the array element VALUE(4) for the case of polarization was omitted from the ASP II for no apparent reason. Implementation of this equation in BEMEXC is questionable and should be reviewed for correctness.

For the eighth discrepancy, equations in BEMOUT for the adjustment of the difference channel voltages to account for cross-polarization bleedover don't make sense. This is because the units don't match with the final result. These equations should also be reviewed for correctness and modified as necessary.

The ninth discrepancy appears to be just documentation errors that can be corrected by rewriting Design Element 5-9 for Continuous Noise Amplitude Calculation in the ASP II.

A discrepancy found during software testing resulted in a program fatal error caused by continuous noise jamming against a fuze radar. The subroutine BEMNZ needs to be modified to account for the case of fuze radar jamming. Otherwise, the concept of jamming a fuze radar using Onboard noise ECM techniques should be removed from this FE along with the subroutine BEMFVL.

Another discrepancy found during software testing was the result of the ALQ-184 jammer data file, ALQ184.DAT which is used for implementation of the Swept Spot Noise ECM technique (among others), being incorrect. This file has the azimuth angles listed backwards in the tables containing the jammer antenna gain patterns for all ECM techniques. This error causes the subroutine TLU2, which looks up values from 2-D tables, to return an incorrect value. The jammer antenna gain tables for all techniques in this file need to be modified to have the antenna azimuth angles increasing from left-to-right across the page, not decreasing as they presently are. The antenna pattern tables in the data files for the other jammers do not have this problem and are correctly implemented.

2.5.5.2 Code Quality and Internal Documentation

The quality of the code for the Onboard Noise ECM FE in ESAMS 2.7 is generally good. Nonetheless, variable declarations are missing from the subroutine BEMANT and should be added. Internal documentation is good for the subroutine BEMNZ. However, the subroutines BEMGRM, BEMSEN, BEMTVL, BEMSVL, BEMFVL, BEMANT, BEMEXC, BEMOUT, BEMSET, and ECMINI have numerous variable description errors and some contain unnecessary INCLUDE statements for common blocks as well. In addition, the comments preceding the array element RADVLU(2) in the subroutines BEMTVL, BEMSVL, and BEMFVL are wrong and need to be corrected. Most of these subroutines do not have completely documented information regarding the subroutine's author, version #, abstract, and purpose/technical description as well.

2.5.5.3 External Documentation

The external documentation for ESAMS 2.7 is good for the subjects discussed in the ASP II; however, a programmer's manual would be useful to describe the software implementation of the theory used to develop ESAMS. Other than choosing which jammer/technique to use, there is no direct user interface to the Onboard Noise ECM FE, therefore, it is not discussed in the *User's Manual* [5]. The *ECM Manual* [6] contains an adequate, although upper level explanation of Onboard Noise ECM methodology. Discrepancies between the source code and the ASP II exist in several subroutines and are discussed in Table 2.5-4.

As mentioned in Section 2.5.3, substantial flowchart errors were found for the subroutine BEMGRM and minor ones were found for most of the others. An effort should be made to update the flowcharts for the subroutines in this FE in a timely manner whenever the code has been modified.

The discussion of look-up tables in the *Advanced User's Manual* [7] should state that azimuth angles for 2-D tables need to increase from left to right across the page and that elevation angles should increase from top to bottom as well. A similar situation probably exists for 3-D lookup tables and should be documented accordingly.